#10

Official

6-17-03

Due Date: June 18, 2003

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of:	
Inventor: Alexander Thoemmes et al.	Examiner: Yang, Ryan R.
Serial #: 09/256,896	Group Art Unit: 2672
Filed: February 24, 1999	Appeal No.: _____
Title: ACQUIRING AND UNACQUIRING ALIGNMENT AND EXTENSION POINTS	

BRIEF OF APPELLANTS

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

In accordance with 37 CFR §1.192, Appellants hereby submit the Appellants' Brief on Appeal from the final rejection in the above-identified application, in triplicate, as set forth in the Office Action dated January 21, 2003.

Please charge the amount of $320 to cover the required fee for filing this Appeal Brief as set forth under 37 CFR §1.17(c) to Deposit Account No. 50-0494 of Gates & Cooper LLP. Also, please charge any additional fees or credit any overpayments to Deposit Account No. 50-0494 of Gates & Cooper LLP.

I. REAL PARTY IN INTEREST

The real party in interest is Autodesk, Inc., the assignee of the present application.

II. RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences for the above-referenced patent application.

07/16/2003 VBUTLER 00000003 500494 09256896

01 FC:1402 320.00 DA

III. STATUS OF CLAIMS

Claims 1-38 are pending in the application.

Claims 36 and 38 were rejected under 35 U.S.C. §102(b) as being anticipated by Venolia (5,463,722).

Claims 1, 2, 6-11, 13, 14, 18-22, 24, 25, 29-33, 35, and 37 were rejected under 35 U.S.C. §103(a) as being unpatentable over Venolia (5,463,722) in view of Kimble (6,031,531).

Claims 3-5, 12, 15-17, 23, 26-28, and 34 were rejected under 35 U.S.C. §103(a) as being unpatentable over Venolia and Kimble as applied to claim 1, and further in view of Newell et al. (5,123,087).

All of the above rejections are being appealed.

IV. STATUS OF AMENDMENTS

No amendments to the claims have been made subsequent to the final Office Action.

V. SUMMARY OF THE INVENTION

Independent claims 1, 13, 24, 35, and 36 are generally directed to operations in a computer drawing program. Specifically, the claims are directed to acquiring/unacquiring a data point. The claims provide for a data point of interest that exists on a drawing object. As cited in the dependent claims, such a data point may be an endpoint of the drawing object, midpoint of the drawing object, a node of the drawing object, a closest quadrant point on the drawing object, an insertion point on the drawing object, a point on a line tangent to the drawing object, or a point on a line that forms a normal from the drawing object (e.g., see claims 5, 17, and 28). A cursor is moved near the data point on the object. Once the cursor remains near the data point for a defined period of time (i.e. an acquisition pause time), the data point is acquired.

In view of the above limitations, the claims are broken up as follows:

A. Data Point Acquiring with Acquisition Pause Time

As described above, independent claims 1, 13, and 24 are method, apparatus (with means plus function language), and program storage device claims (with instructions that perform a

method) directed towards acquiring a data point on an object when a cursor remains near the data point for a defined period of time.

Dependent claims 2-12, 14-23, and 25-35 are all dependent on independent claims 1, 13, and 24 respectively.

Dependent claims 2, 14, and 25 provide that the pause time is user-selectable.

Dependent claims 3, 15, and 26 provide that the drawing object that contains the data point of interest is a linear entity.

Dependent claims 4, 16, and 27 depend on claims 3, 15, and 26 respectively. These dependent claims add the further step of extending the linear entity by accepting a command to move the cursor away from the data point.

Dependent claims 5, 17, and 28 provide for a group for the various types of data points. Specifically, as described above, a data point may be an endpoint, a midpoint, a node, a closest quadrant point, an insertion point, a point on a line tangent to the object, and/or a point on a line that forms a normal from the object.

Dependent claims 6, 18, and 29 add the further limitation that the cursor remains within a particular distance (referred to as an acquisition distance) of the data point during the acquiring process.

Dependent claims 7, 19, and 30 depend on claims 6, 18, and 29 respectively. These dependent claims provide the ability to determine the acquisition distance based on a particular parameter. More specifically, the particular parameter may be a magnification of a view of the object and/or an object type.

Dependent claims 8, 20, and 31 provide the further limitation of annotating the data point once it has been acquired. This annotation takes the form of an acquisition indicator.

Dependent claims 9, 21, and 32 add the further limitation of unacquiring the data point. Such an unacquiring occurs after the cursor remains near the data point for a particular amount of time (referred to as an unacquisition pause time).

Dependent claims 10, 22, and 33 provide further capability to unacquire the acquired data point. In this regard, the cursor is moved away from the acquired data point and then near the acquired data point again. Once the cursor remains near the acquired data point again for a

particular amount of time (referred to as the unacquisition pause time), the acquired data point is unacquired.

Dependent claim 11 depends on claim 10 and specifies that the unacquisition pause time is different from the acquisition pause time.

Dependent claims 12, 23, and 34 provide the ability to align two objects. The cursor is moved towards a second data point on a second object and the second data point is acquired similar to that of acquiring the first data point (i.e., after an acquisition pause time). Once the second data point is acquired, the first object (containing the first data point) and the second object (containing the second data point) are aligned.

B. Data Point Unacquiring

Independent claim 35 is a method claim directed towards unacquiring an already acquired data point. The data point is unacquired after the cursor remains near an acquired data point for a defined period of time (referred to as the unacquisition pause time).

C. Data Point Acquiring with a Modifier Command

Independent claim 36 is a method claim directed towards acquiring a data point. Specifically, independent claim 36 provides for acquiring a data point of interest on an object in a drawing only when a modifier command has been accepted. For example, in accordance with dependent claim 38, the modifier command may comprise depressing a keyboard key and the point is only selected when the keyboard key is depressed. Further, dependent claim 37 adds acquiring the data point after the cursor remains near the data point for an acquisition pause time.

VI. ISSUES PRESENTED FOR REVIEW

Whether claims 36 and 38 are anticipated under 35 U.S.C. §102(b) by Venolia (5,463,722).

Whether claims 1, 2, 6-11, 13, 14, 18-22, 24, 25, 29-33, 35, and 37 are obvious under 35 U.S.C. §103(a) over Venolia (5,463,722) in view of Kimble (6,031,531).

Whether claims 3-5, 12, 15-17, 23, 26-28, and 34 are obvious under 35 U.S.C. §103(a) as over Venolia and Kimble, and further in view of Newell et al. (5,123,087).

VII. GROUPING OF CLAIMS

The rejected claims do not stand or fall together.

VIII. ARGUMENTS

A. Claims 36 and 38 are not anticipated by Venolia (5,463,722).

In paragraphs (4)-(5) of the final Office Action, claims 36 and 38 were rejected under 35 U.S.C. §102(b) as being anticipated by Venolia, U.S. Patent No. 5,463,722 (Venolia).

Specifically, claim 36 was rejected as follows:

> As per claim 36, Kimble discloses a method of acquiring a data point of interest on a drawing object, comprising the steps of:
> accepting a modifier command ("keyboard commands or menu selections for creating and breaking such multiple object alignments", column 22, line 9-11); and
> acquiring the data point of interest on a drawing object in a computer-implemented drawing program after a command is received to move a cursor near the data point, wherein the data point is not acquired without the modifier command (without pressing down the keyboard, the objects are not aligned).

Applicants traverse the above rejections for one or more of the following reasons:

1. *Venolia fails to teach disclose, or suggest accepting a modifier command; and*
2. *Venolia fails to teach disclose, or suggest acquiring a data point of interest on a drawing object when a cursor is moved near the data point, wherein the data point is not acquired without the modifier command.*

As described above, independent claim 36 provides for acquiring a data point of interest on an object in a drawing only when a modifier command has been accepted. For example, in accordance with claim 38, the modifier command may comprise depressing a keyboard key and the point is only selected when the keyboard key is depressed. Additionally, the data point of interest is acquired after a cursor is moved near the data point.

The final Office Action relies on Venolia col. 22, lines 9-11 to teach all of the steps of claim 36. Col. 22, lines 9-11 of Venolia provides:

> This capability would preferably include an interface selection such as, for example, keyboard commands or menu selections for creating and breaking such multiple object alignments.

Claims 36-38 do not provide or even allude to aligning objects. Instead, the claims specifically provide for selecting a data point of interest on a drawing object. Nowhere in the cited text of Venolia is there any reference to a particular data point of interest on an object. Further, Venolia completely fails to describe selecting such a particular data point of interest when a modifier command, such as the depression of a keyboard key, has been accepted.

The Examiner may be correct in stating that Venolia provides that without pressing down the keyboard, the objects are not aligned. However, the present claims are not directed towards object alignment but point acquisition. Aligning objects is a completely different chatacteristic/process/capability of an object than that of selecting a point of interest on an object.

In response to the above arguments, the Advisory Action provides:

> As per claims 36 and 38, Applicant alleges Venolia fails to suggest a data point of interest on a drawing object and fails to teach acquiring a data point of interest on a drawing object. In reply, Examiner notes that Venolia does indeed provide point of interest by providing attractive vertex designated as Q (column 12, line 14) and pulling a point P to another point Q is an acquiring process (column 12, line 16). As for selecting a data point of interest, it is not part of the claim limitation. Applicant also argues that acquiring is a different process from aligning. However, since the title of the invention is "Acquiring and unacquiring alignment and extension points", it suggests acquiring is part of the alignment process. Indeed, without acquiring a point, or points, alignment cannot be accomplished.

Applicant respectfully traverses the Examiner's reply. The Advisory Action suggests that Venolia's attractive vertex Q is acquired when another point P is pulled towards it. However, Applicants submit that point Q is never acquired in Venolia. Col. 12, lines 5-18 provides:

> ALIGNING TWO VERTICES
>
> In a preferred embodiment, when the user drags a vertex of a displayed object towards the vertex of another object displayed in a scene, based on the model of magnetic attraction, the attraction between the two objects becomes stronger as the objects move closer together. FIG. 3 illustrates the effect in a preferred embodiment. The vertex that is being dragged is designated P, and the attractive vertex that is attracting P in the scene is designated as Q. The cursor specifies a position, A, for the dragged vertex. If P were pulled into complete alignment with Q, its position would be B. The vertex P is displayed at C, a position that is influenced by both the cursor position and P's attraction to Q.

As described in col. 12, point P is merely moved towards Q and aligned. Venolia fails to even mention acquiring point Q whatsoever. Instead, P is merely aligned with Q. Accordingly, contrary to the assertion in the Advisory Action, pulling a first point towards another second point is not equivalent to acquiring the second point.

The Advisory Action also asserts that since the title of the invention mentions acquiring alignment and extension points, the process of acquiring is therefore part of the alignment process. In addition, the Advisory Action asserts that without acquiring a point or points, alignment cannot be accomplished. Such an assertion is flawed for various reasons.

Firstly, it is improper to read any limitations from a title of the invention into the claims.

Secondly, an object may be used for alignment and not a point or points. In this regard, contrary to the assertion in the Advisory Action, an object and not a point may be acquired as part of the alignment process.

Thirdly, regardless of whether the claims provide for aligning an object, the prior art must teach the steps of the claims. If the prior art provides that acquiring a point is part of the alignment process, then it may read on the invention. The Advisory Action suggests that as part of our invention, alignment is part of the acquiring process. While Applicants disagree with such a suggestion, its veracity and such a teaching is not relevant. Regardless of what Applicant's invention does, the prior art must teach the elements as claimed. In this regard, Venolia teaches aligning two vertices. In Venolia, acquiring a data point of interest on a drawing object is not part of the aligning process. Accordingly, Venolia fails to teach, describe, or suggest the invention as claimed.

In addition to the above, the claims recite not acquiring a data point without a modifier command. As with the other claim elements, Venolia col. 22, lines 9-11 is relied upon. However, the context of these lines must be understood. Venolia col. 22, lines 4-11 provides:

> In an even further alternative embodiment, once the user aligns two objects, the system of the present invention could maintain the alignment and moves the two objects as a single object or group as indicated by example (d) of FIG. 10. This capability would preferably include an interface selection such as, for example, keyboard commands or menu selections for creating and breaking such multiple object alignments.

This portion of Venolia recites using keyboard commands or menu selections for creating and breaking multiple object alignments. However, there is no restriction within this portion of Venolia or throughout Venolia that provides that the alignment cannot occur without using the keyboard or menu commands. Thus, Venolia does not require the use of the keyboard or menu command as part of the "alignment" process. In addition, as stated above, the alignment process is clearly distinguishable from acquiring a data point. In this regard, using a keyboard command to align multiple objects is not related to and does not describe acquiring a data point.

Accordingly, Applicants submit that claims 36-38 are allowable over Venolia.

B. Independent claims 1, 13, 24, and 35 are Patentable Over Venolia (5,463,722) in View of Kimble (6,031,531).

In paragraphs (7)-(21) of the Office Action, claims 1, 2, 6-11, 13, 14, 18-22, 24, 25, 29-33, 35, and 37 were rejected under 35 U.S.C. §103(a) as being unpatentable over Venolia in view of Kimble, U.S. Patent No. 6,031,531 (Kimble). In paragraph (22) of the Office Action, claims 3-5, 12, 15-17, 23, 26-28, and 34 were rejected under Venolia and Kimble as applied to claim 1, and further in view of Newell et al., U.S. Patent No. 5,123,087 (Newell).

Specifically, independent claims 1, 13, 24, and 35 were rejected as follows:

> As per claim 1,Venolia discloses a method of acquiring a data point of interest on a drawing object, comprising the steps of:
> accepting a command to move a cursor near the data point of interest on the drawing object in a computer-implemented drawing program (Figure 3); and
> acquiring the data point after the cursor remains near the data point (Figure 3).
> Venolia discloses a method of acquiring a cursor when moved within a distance. It is noted that Venolia does not explicitly disclose acquiring the data point after the cursor remains near the data point for an acquisition pause time, however, this is known in the art as taught by Kimble. Kimble discloses acquiring a cursor after the cursor remains nears an object after a period to time (Figure 7 164, "By dwelling on the icon/object (i.e., by not utilizing a switch or moving the cursor), the function associated with the icon/object upon which the cursor is "dwelling" is automatically activated", column 9, line 34-37).
>
> As per claim 13, Venolia discloses an apparatus for acquiring a data point of interest on a drawing object, comprising:
> means for accepting a command to move a cursor near the data point of the drawing object in a computer-implemented drawing program (Figure 1 1610).
>
> As per claim 24, Venolia discloses a program storage device (Figure 1 1616), readable by a computer, tangibly embodying at least one program of instructions executable by the computer in a drawing program to perform method steps of acquiring a data point of interest on a drawing object (Figure 1 1610), the method comprising the steps of:
> accepting a command to move a cursor near the data point of interest on the drawing object (Figure 1 1610).
>
> As per claim 35, Venolia discloses a method of unacquiring an acquired data point, comprising the steps of:
> accepting a command to move a cursor near the acquired data point of a drawing object in a computer-implemented drawing program (Figure 3).

Applicants traverse the above rejections for one or more of the following reasons:

1. Neither Venolia, Kimble, nor Newell teach, disclose or suggest acquiring a data point of interest on a drawing object; and

2. Neither Venolia, Kimble, nor Newell teach, disclose or suggest acquiring a data point after a cursor remains near the data point for an acquisition pause time.

As described above, independent claims 1, 13, 24, and 35 are generally directed to operations in a computer drawing program. Specifically, the claims are directed to acquiring/unacquiring a data point. The claims provide for a data point of interest that exists on a drawing object. As cited in the dependent claims, such a data point may be an endpoint of the drawing object, midpoint of the drawing object, a node of the drawing object, a closest quadrant point on the drawing object, an insertion point on the drawing object, a point on a line tangent to the drawing object, or a point on a line that forms a normal from the drawing object (e.g., see claims 5, 17, and 28). A cursor is moved near the data point on the object. Once the cursor remains near the data point for a defined period of time (i.e. an acquisition pause time), the data point is acquired.

The cited references do not teach nor suggest these various elements of Applicants' independent claims.

Venolia merely describes a method and apparatus for automatic alignment of manipulated objects in two-dimensional and three-dimensional graphic space.

The second step of the claims provides for "acquiring the data point after the cursor remains near the data point for an acquisition pause time." Applicants note that the data point is further limited by the first claim element as a data point of interest on a drawing object. To teach, this element, Venolia's Figure 3 is relied upon. Col. 12, lines 6-18 describes Fig. 3:

> In a preferred embodiment, when the user drags a vertex of a displayed object towards the vertex of another object displayed in a scene, based on the model of magnetic attraction, the attraction between the two objects becomes stronger as the objects move closer together. FIG. 3 illustrates the effect in a preferred embodiment. The vertex that is being dragged is designated P, and the attractive vertex that is attracting P in the scene is designated as Q. The cursor specifies a position, A, for the dragged vertex. If P were pulled into complete alignment with Q, its position would be B. The vertex P is displayed at C, a position that is influenced by both the cursor position and P's attraction to Q.

As indicated in this cited portion, a vertex of one object is merely being dragged towards another object. The cited portion provides for moving one vertex (referred to herein as the first vertex) using the cursor towards another vertex (referred to herein as the second vertex). Firstly,

Venolia does not provide for moving the cursor near the first vertex and then acquiring the vertex after the cursor remains near the point for a pause time. Thus, the use of the first vertex in Venolia fails to teach the invention as claimed. Additionally, Venolia's second vertex is not acquired. Instead, the second vertex is merely used for alignment. Further, there is no use of a "pause time" as claimed with respect to the second vertex.

In fact, the Office Action provides that Venolia "does not explicitly disclose acquiring the data point after the cursor remains near the data point for an acquisition pause time". Instead, the Office Action relies on Kimble to teach this element of the claim. However, as stated in the prior Office Action, Kimble's dwelling merely activates a function associated with an object/icon and does not select/acquire a particular point on a drawing object. Contrary to the assertion in the Office Action, Kimble does not disclose "acquiring a cursor when the cursor is within a distance of an object after a period of time in order to easily access the object".

As cited throughout Kimble, the "dwelling" is merely used to activate a function associated with the icon/object (see col. 9, lines 34-37). In an embodiment described in Kimble, the cursor snaps to an icon if the user doesn't move the cursor away from the icon after a time interval (see col. 10, lines 31-35). Thus, Kimble merely describes the activation or snapping to an icon. In this regard, Kimble fails to describe a drawing object whatsoever. Kimble's icon/object is not equivalent to a drawing object in a computer-implemented drawing program. It should be noted that Kimble is merely directed towards assisting physically challenged users with icon selection in a graphical interface of a computer program. Kimble does not teach or describe, explicitly or implicitly, a drawing program or the use in/of a drawing program.

Additionally, the claims provide and cite a data point of interest on a drawing object. Kimble completely fails to describe a data point. Even assuming that Kimble's icon/object is equivalent to the present invention's drawing object (although Applicants traverse such an assertion), Kimble does not describe any data point on the icon/object. Instead, Kimble merely describes an entire icon/object without describing aspects of the object.

Further, the claims provide for acquiring the data point of interest on the drawing object. Kimble merely describes the activation of a function associated with the complete icon or a cursor

snapping to the icon location. Acquiring a particular data point on a drawing object is not even remotely similar to activating a function of an icon or snapping to a location of an icon.

The Office Action provides that Kimble and Venolia can be combined because they both disclose a method of acquiring a cursor. However, it should be noted that the claims do not provide for acquiring a "cursor" but acquiring a data point of interest on a drawing object after a cursor remains near the data point. Kimble is merely directed towards assisting physically challenged users with icon selection in a graphical interface of a computer program. Kimble does not teach or describe, explicitly or implicitly, a drawing program or the use in/of a drawing program. Further, Venolia is directed towards alignment of objects. The alignment of objects is completely different from assisting physically challenged users by activating a function of an icon after a cursor remains near the icon. In this regard, the two fields of invention are non-analogous and there is no suggestion to combine the references, either explicitly or implicitly.

In response to the above arguments, the Advisory Action replies:

> In reply, Examiner notes that Venolia does indeed teach acquiring a point (see above argument for claims 36 and 38) and the Examiner uses the Kimble reference to meet the pause time limitation.

Accordingly, in response to prior arguments, the Advisory Action now merely utilizes Kimble to teach a limitation of a pause time as claimed. However, as stated above, the pause time is closely tied together with the effect of the pause time as claimed. Using an abstract reference such as Kimble to teach a pause time that has no other relation to the claims or field of the invention whatsoever is improper.

Applicants again assert that there is no motivation or suggestion to combine the references. The MPEP §706.02(j) specifically provides that "there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings." In this regard, neither Venolia nor Kimble contain any motivation or suggestion to combine with each other. As described above, Kimble is directed towards a "technology which assists physically challenged users in efficiently utilizing graphical user interface applications" (see col. 1, lines 12-15). However, Venolia relates to "computer graphics and in particular to an alignment field gradient exerted by objects surrounding an object displayed in a two-dimensional (2D) or three-dimensional (3D) graphical display space

appearing on a computer screen" (see col. 1, lines 7-12). There is no suggestion, implicit or explicit to combine Venolia's alignment process with Kimble's physically challenged user technology.

In addition to the above, like Venolia and Kimble, Newell fails to teach, describe, or suggest various claim elements.

Further, the various elements of Applicants' claimed invention together provide operational advantages over the systems disclosed in Venolia, Kimble, and Newell. In addition, Applicants' invention solves problems not recognized by Venolia, Kimble, and Newell.

Thus, Applicants submit that independent claims 1, 13, 24, and 35 are allowable over Venolia, Kimble, and Newell.

C. Dependent Claims 2, 14, and 25 Are Not Separately Argued.

D. Dependent claims 3, 4, 15, 16, 26, and 27 Are Not Separately Argued.

E. Dependent claims 5, 17, and 28 Are Not Separately Argued

F. Dependent claims 6, 18, and 29 Are Not Separately Argued.

G. Dependent Claims 7, 19, and 30 Are Patentable over Venolia and Kimble.

As described above, dependent claims 7, 19, and 30 depend on claims 6, 18, and 29 respectively. These dependent claims provide the ability to determine the acquisition distance based on a particular parameter. More specifically, the particular parameter may be a magnification of a view of the object and/or an object type. To reject these claims the final Office Action states that the magnification of a view of the object and object type parameters are taught by Kimble col. 9, lines 24-26 which provides:

> ...The amount of cursor movement necessary to trigger the hop can be adjustable by the user when configuring this particular feature (i.e., configuring a particular threshold)...

However, such a citation in Kimble completely fails to describe or even remotely suggest a particular type of parameter (as claimed) to be used to determine an acquisition distance. The claims specifically recite a parameter that is selected from a group comprising a magnification of a view of

the object and an object type. Kimble fails to recite either of these parameters. Adjusting a hop distance or configuring a particular threshold does not describe, implicitly or explicitly, either a magnification of a view of the object or an object type. Accordingly, Kimble fails to teach the invention as claimed.

H. Dependent claims 8, 20, and 31 Are Patentable over Venolia and Kimble

As described above, claims 8, 20, and 31 provide the limitation of annotating the data point once it has been acquired. This annotation takes the form of an acquisition indicator. In rejecting these claims, the Office Action relies on Kimble col. 7, lines 33-37. This portion of Kimble refers to a "magnetized" area that surrounds an icon. However, the context of this paragraph must be examined. Col. 7, lines 19-43 provides as follows:

> FIG. 6(a) illustrates a pictorial representation of an icon 70 and an associated icon domain 65, in accordance with a preferred embodiment of the present invention. **Icon 70 is essentially a "magnetized" graphical object and may be displayed within a window such as window 50 of FIG. 3 and FIG. 4. The term "magnetized" is utilized figuratively herein to refer to a property of icons or objects utilized in association with a preferred embodiment of the present invention.** Cursor object 63, depicted in FIG. 6(a), is analogous to cursor object 63 of FIG. 3 and FIG. 4. FIG. 6(b) depicts a pictorial representation illustrating the movement of cursor object 63 depicted in FIG. 6(a) through icon domain 65, in accordance with a preferred embodiment of the present invention. Arrow 68 in FIG. 6(b) specifically illustrates the movement of cursor object 63. The icon is "magnetized" such that an area outlined by icon domain 65, with a diameter of perhaps two inches (i.e., depending upon a particular implementation of a preferred embodiment of the present invention), surrounds icon 70. When cursor object 63 is moved into the area outlined by icon domain 65, cursor object 63 is immediately "snapped" to the center of icon 70, which the user may then activate. In response to a particular user input, such as a particular keystroke or combination of keystrokes, the "magnetic" property (i.e., icon domain), may be enabled globally (i.e., for all objects). (EMPHASIS ADDED)

As illustrated in the above portion of Kimble, the "magnetized" area is a figurative term that merely comprises a property of icons or objects in Kimble. Further, the object is actually displayed as illustrated in FIG. 3 and FIG. 4 (i.e., without any surrounding area). Accordingly, the dashed line illustrated in FIG. 6a is not what is actually displayed in the program. In fact, as just described, Kimble specifically provides that dashed line is not displayed.

Accordingly, Kimble fails to provide, teach, disclose, or suggest an annotation of a data point (as claimed).

I. Dependent claims 9, 10, 21, 22, 32, and 33 Are Patentable Over Venolia and Kimble.

As described above, claims 9, 10, 21, 22, 32, and 33 provide for unacquiring the data point. Such an unacquiring occurs after the cursor remains near the data point for a particular amount of time (referred to as an unacquisition pause time). To teach these claims, the Office Action relies on Kimble's teaching of a demagnetization in col. 10, lines 10-11. Applicants traverse these assertions for the reasons stated above and in addition for those stated herein.

Nowhere in Kimble is there any mention of demagnetizing based on the cursor remaining near Kimble's icons for a particular amount of time. Instead, Kimble's object merely remains demagnetized for a specified time interval. The amount of time an object is demagnetized is not equivalent to demagnetizing an object after a cursor remains near an object for a particular amount of time. In other words, the present claims provide for moving a cursor into an area and timing how long the cursor remains there in order to determine if a data point should be unacquired. Such a process is contrary to demagnetizing an object and then keeping the object demagnetized for a specified time interval. In this regard, while the present claims provide a precondition for determining if a data point should be unacquired, Kimble teaches a post-demagnetizing time period. Such a teaching does not render these dependent claims obvious.

J. Dependent Claim 11 is Patentable Over Venolia and Kimble.

As described above, dependent claim 11 depends on claim 10 and specifies that the unacquisition pause time is different from the acquisition pause time. To teach this claim element, the final Office Action provides that having different values is inherent to differentiate one time from the other. Applicants traverse such an assertion. A cursor may be placed near a data point for a first period of time after which the data point is acquired. Additionally, if the cursor is moved away from the data point and then moved back for the same amount of time (i.e., the first period of time), the data point may be unacquired. Such a possibility clearly establishes that it is not inherent that the times be different. Nonetheless, the present claims provide that the time period for acquiring and unacquiring are different. Such a limitation is not taught or suggested by the prior art. Nor is such a limitation inherent.

K. Dependent claims 12, 23, and 34 Are Not Separately Argued.

L. Conclusion

In light of the above arguments, Appellants respectfully submit that the cited references do not anticipate nor render obvious the claimed invention. More specifically, Appellants' claims recite novel physical features which patentably distinguish over any and all references under 35 U.S.C. §§ 102 and 103. As a result, a decision by the Board of Patent Appeals and Interferences reversing the Examiner and directing allowance of the pending claims in the subject application is respectfully solicited.

Respectfully submitted,

GATES & COOPER LLP
Attorneys for Applicant(s)
Howard Hughes Center
6701 Center Drive West, Suite 1050
Los Angeles, California 90045
(310) 641-8797

Date: June 17, 2003

By: _____
Name: Jason S. Feldmar
Reg. No.: 39,187

JSF/amb
G&C 30566.60-US-01

APPENDIX

1. (PREVIOUSLY AMENDED) A method of acquiring a data point of interest on a drawing object, comprising the steps of:

accepting a command to move a cursor near the data point of interest on the drawing object in a computer-implemented drawing program; and

acquiring the data point after the cursor remains near the data point for an acquisition pause time.

2. (ORIGINAL) The method of claim 1, wherein the pause time is user-selectable.

3. (ORIGINAL) The method of claim 1, wherein the object is a linear entity.

4. (ORIGINAL) The method of claim 3, further comprising the step of accepting a command to move the cursor away from the data point to extend the linear entity.

5. (ORIGINAL) The method of claim 1, wherein the data point is selected from a group comprising:

an endpoint;

a midpoint;

a node;

a closest quadrant point;

an insertion point;

a point on a line tangent to the object; and

a point on a line that forms a normal from the object.

6. (ORIGINAL) The method of claim 1, wherein the step of acquiring the data point after the cursor remains near the data point for an acquisition pause time comprises the step of acquiring the data point after the cursor remains within an acquisition distance of the data point for an acquisition pause time.

7. (ORIGINAL) The method of claim 6, wherein the acquisition distance is determined according to a parameter selected from a group comprising

magnification of a view of the object; and

an object type.

8. (ORIGINAL) The method of claim 1, further comprising the step of annotating the acquired data point with an acquisition indicator.

9. (ORIGINAL) The method of claim 1, further comprising the step of unacquiring the data point after the cursor remains near the acquired data point for an unacquisition pause time.

10. (ORIGINAL) The method of claim 1, further comprising the steps of:

accepting a command to move the cursor away from near the data point;

accepting a command to move the cursor near the data point; and

unacquiring the data point after the cursor remains near the data point for the unacquisition pause time.

11. (ORIGINAL) The method of claim 10, wherein the unacquisition pause time is a different value than the acquisition pause time.

12. (ORIGINAL) The method of claim 1, further comprising the steps of:

accepting a command to move the cursor near a second data point on a second object;

acquiring the second data point after the cursor remains near the second data point for the acquisition pause time; and

aligning the first object and the second object according to the acquired first data point and the acquired second data point.

13. (PREVIOUSLY AMENDED) An apparatus for acquiring a data point of interest on a drawing object, comprising:

means for accepting a command to move a cursor near the data point of the drawing object in a computer-implemented drawing program; and

means for acquiring the data point after the cursor remains near the data point for an acquisition pause time.

14. (ORIGINAL) The apparatus of claim 13, wherein the pause time is user-selectable.

15. (ORIGINAL) The apparatus of claim 13, wherein the object is a linear entity.

16. (ORIGINAL) The apparatus of claim 15, further comprising means for accepting a command to move the cursor away from the data point to extend the linear entity.

17. (ORIGINAL) The apparatus of claim 13, wherein the data point is selected from the group comprising:

an endpoint;

a midpoint;

a node;

a closest quadrant point;

an insertion point;

a point on a line tangent to the object; and

a point on a line that forms a normal from the object.

18. (ORIGINAL) The apparatus of claim 13, wherein the means for acquiring the data point after the cursor remains near the data point for an acquisition pause time comprises the step of acquiring the data point after the cursor remains within an acquisition distance of the data point for an acquisition pause time.

19. (ORIGINAL) The apparatus of claim 18, wherein the acquisition distance is determined according to a parameter selected from a group comprising:

magnification of a view of the object; and

an object type.

20. (ORIGINAL) The apparatus of claim 13, further comprising means for annotating the acquired data point with an acquisition indicator.

21. (ORIGINAL) The apparatus of claim 13, further comprising means for unacquiring the data point after the cursor remains near the acquired data point for an unacquisition pause time.

22. (ORIGINAL) The apparatus of claim 13, further comprising:

means for accepting a command to move the cursor away from near the data point;

means for accepting a command to move the cursor near the data point; and

means for unacquiring the data point after the cursor remains near the data point for the unacquisition pause time.

23. (ORIGINAL) The apparatus of claim 13, further comprising:

means for accepting a command to move the cursor near a second data point on a second object;

means for acquiring the second data point after the cursor remains near the second data point for the acquisition pause time; and

means for aligning the first object and the second object according to the acquired first data point and the acquired second data point.

24. (PREVIOUSLY AMENDED) A program storage device, readable by a computer, tangibly embodying at least one program of instructions executable by the computer in a drawing program to perform method steps of acquiring a data point of interest on a drawing object, the method comprising the steps of:

accepting a command to move a cursor near the data point of interest on the drawing object; and

acquiring the data point after the cursor remains near the data point for an acquisition pause time.

25. (ORIGINAL) The program storage device of claim 24, wherein the pause time is user-selectable.

26. (ORIGINAL) The program storage device of claim 24, wherein the object is a linear entity.

27. (ORIGINAL) The program storage device of claim 26, wherein the method steps further comprise the step of accepting a command to move the cursor away from the data point to extend the linear entity.

28. (ORIGINAL) The program storage device of claim 24, wherein the data point is selected from the group comprising:

an endpoint;

a midpoint;

a node;

a closest quadrant point;

an insertion point;

a point on a line tangent to the object; and

a point on a line that forms a normal from the object.

29. (ORIGINAL) The program storage device of claim 24, wherein the method step of acquiring the data point after the cursor remains near the data point for an acquisition pause time comprises the step of acquiring the data point after the cursor remains within an acquisition distance of the data point for an acquisition pause time.

30. (ORIGINAL) The program storage device of claim 29, wherein the acquisition distance is determined according to a parameter selected from a group comprising:

magnification of a view of the object; and

an object type.

31. (ORIGINAL) The program storage device of claim 24, wherein the method steps further comprise the method step of annotating the acquired data point with an acquisition indicator.

32. (ORIGINAL) The program storage device of claim 24, wherein the method steps further comprise the step of unacquiring the data point after the cursor remains near the acquired data point for an unacquisition pause time.

33. (ORIGINAL) The program storage device of claim 24, wherein the method steps further comprise the steps of:

accepting a command to move the cursor away from near the data point;

accepting a command to move the cursor near the data point; and

unacquiring the data point after the cursor remains near the data point for the unacquisition pause time.

34. (ORIGINAL) The program storage device of claim 24, wherein the method steps further comprise the steps of:

accepting a command to move the cursor near a second data point on a second object;

acquiring the second data point after the cursor remains near the second data point for the acquisition pause time; and

aligning the first object and the second object according to the acquired first data point and the acquired second data point.

35. (PREVIOUSLY AMENDED) A method of unacquiring an acquired data point, comprising the steps of:

accepting a command to move a cursor near the acquired data point of a drawing object in a computer-implemented drawing program; and

unacquiring the data point after the cursor remains near the acquired data point for an unacquisition pause time.

36. (PREVIOUSLY AMENDED) A method of acquiring a data point of interest on a drawing object, comprising the steps of:

accepting a modifier command; and

acquiring the data point of interest on a drawing object in a computer-implemented drawing program after a command is received to move a cursor near the data point, wherein the data point is not acquired without the modifier command.

37. (ORIGINAL) The method of claim 36, wherein the data point is acquired after the cursor remains near the data point for an acquisition pause time.

38. (PREVIOUSLY ADDED) The method of claim 36 wherein the modifier command comprises the depression of a keyboard key.